**COLEGIO REALENGO**

**NICKOLAS DOS SANTOS CREMASCO**

# PYTHON A LINGUAGEM QUE MUDOU O MUNDO

Realengo - RJ
2024

**AGRADECIMENTOS**

Inicialmente queria agradecer a todos os meus professores do núcleo técnico que me deram toda a orientação para criar este artigo que foi feito com muita força de vontade e ímpeto até o fim.

Obrigado a todos também que me ajudaram indiretamente, autores de bons livros, amigos, família entre outros.

## RESUMO

Python é uma linguagem de programação chave para toda e qualquer pessoa que queira ingressar na área da programação, além de ser uma linguagem nova, moderna e fácil de aprender tem em suma uma comunidade extremamente ativa que cria e disponibiliza bibliotecas para aumentar ainda mais a variedade de programas possíveis dentro dessa linguagem como aprendizado de máquina, análise de dados e afins que serão aprofundados dentro do artigo.



**Palavras-chave:** Python. Programação. Dados.

**ABSTRACT**

Python is a key programming language for anyone and everyone who wants to enter the programming field. In addition to being a new, modern and easy-to-learn language, it also has na extremely active Community that creates and makes libraries available to further increase the variety of possible programs within this language such as machine learning, data analisys and a lots details that will be aborded along of this article.

**Keywords:** Python. Programming. Data.

## LISTA DE ILUSTRAÇÕES

Fotografia 1 - Guido Van Rossum.....16
Figura 1 – Programa em Python.....17
Figura 2 – Programa em Java.....17
Figura 3 – Tupla.....18
Figura 4 – Lista.....18
Figura 5 – Dicionário.....19
Figura 6 – If, Elif e Else.....19
Figura 7 – For.....20
Figura 8 – While .....21

**SUMÁRIO**

**1 INTRODUÇÃO**.....
2 Guido Van Rossum.....
2.1 Programador desde a infância.....
2.2 Inspirações Para o Desenvolvimento do Python.....
2.3 Evolução Rápida e Continua.....
2.4 Comunidade Extremamente Ativa.....
2.5 Legado De Guido Van Rossum.....
3 Compiladores.....
3.1 Papel Do Compiladores.....
3.2 CPython: Compilação JIT(Just-In-Time).....
4 Exemplos Práticos em Python.....
4.1 Sintaxe Simples, Mas Poderosa.....
4.2 Variáveis – Baús De Informações.....
4.3 Input – Comando De Leitura.....
4.4 Tuplas, Listas e Dicionários – Cadeias De Informações.....
4.4.1 Tuplas.....
4.4.2 Listas.....
4.4.3 Dicionários.....

Comandos De Exceção.....
4.5
Comandos De Repetição.....
4.6
Comando For.....
4.6.1
Comando While.....
4.6.2
5 Bibliotecas Variadas.....
Desenvolvimento Web Com Django.....
5.1
Automações Web Com Selenium.....
5.2
Automações com PyAutoGui.....
5.3
**CONCLUSÃO**.....
**GLOSSÁRIO**.....
**REFERÊNCIAS**.....

# 1 INTRODUÇÃO

Python é uma linguagem de programação que foi idealizada por Guido Van Rossum no ano de 1989 na Holanda país de origem de seu criador, Surgiu como um desafio enfrentado durante os trabalhos de Van Rossum no Instituto de matemática computacional de Amsterdam. Percebeu uma lacuna significativa nas linguagens de programação disponíveis, a excessiva complexidade na qual eram criadas e aprendidas, a maioria dessas eram de baixo nível, mais próximas do hardware, sendo extremamente complexas para serem aprendidas em curto prazo.

Percebeu a demanda por linguagens mais intuitivas e de alto nível, mais próximas do programador, crescendo cada vez mais o que o impulsionou a criar a linguagem Python a revolucionária linguagem que simplificaria conceitos complexos cujo nome foi escolhido por Van Rossum ser fã de uma série de TV chamada Tropa Monty Python(estréia:1969).

Desde sua criação, o Python tem experimentado um crescimento gigante em popularidade, se tornando uma das linguagens mais utilizadas globalmente. Sua versatilidade é bem visível ainda por ser aplicável em diversas áreas, desde desenvolvimento Web, Machine Learning a automações, se tornando a principal escolha para desenvolvedores novatos que desejam aprender os fundamentos da programação.

Neste artigo o objetivo é explorar o que o Python pode oferecer no cenário da programação contemporânea, quais são seus impactos no mundo da informática e por que se tornou uma das maiores e mais populares linguagens do mundo.

# 2 Guido Van Rossum

## 2.1 Programador desde a Infância

Desde o Início da carreira foi um fã notável da computação, tinha um desejo gigante por desvendar mistérios e a área da programação era um prato cheio para Guido que com mais idade decidiu ingressar no Instituto de Matemática Computacional de Amsterdam, onde desafios e mistérios mais complexos foram atravessando seu caminho.

## 2.2 Inspirações Para o Desenvolvimento do Python

Guido sempre foi fã de programação, mas não era um adepto fugaz quando o assunto era a complexidade que as linguagens traziam consigo, e essa foi a faísca que precisava para ter a ideia que mudaria sua vida e de toda a história da programação, imaginou se novatos pudessem ser protegidos da complexidade que linguagens como Java, C, Assembly entre outras poderiam ter para iniciantes e decidiu criar algo novo que englobasse não apenas programadores experientes mas sim como principal foco os iniciantes.

A partir de linguagens anteriores como o **ABC**, uma linguagem focada em pessoas que não eram especializados na área da programação, voltada para o ensino e produtividade com uma sintaxe limpa e legível, e a **Modula-3** que detinha conceitos mais avançados de tratamento de exceções e orientação a objetos foi criada a arquitetura da linguagem **Python** que teve seu lançamento oficial em 1991 e surpreendeu por sua sintaxe limpa e pela capacidade de ser utilizada em várias áreas diferentes.

## 2.3 Evolução Contínua e Rápida

Python despontou rápido no mercado e se estabeleceu como uma das linguagens mais populares do mundo e pouco tempo, sua

aceitação rápida pela comunidade  é testemunha da visão que Guido teve ao criar algo não apenas eficiente, mas também acessível e revolucionário para todos em qualquer nível, dos mais iniciantes até os mais sábios de linguagens de programação

## 2.4 Comunidade Extremamente Ativa

O Python hoje é dono de uma das comunidades mais ativas do mundo na área da programação, Guido continua tendo papel ativo na evolução da linguagem, mas hoje conta com a participação constante de inúmeras comunidades de desenvolvedores que tem como compromisso fiel a melhora continua da linguagem. Tanto que hoje existem bibliotecas para Desenvolvimentos Web como Django e Flask, Automações de tarefas como Selenium, PyAutoGui, BeatifulSoap entre outras e até para Machine Learning ou Desenvolvimento de IA como Tensor Flow ou PyTorch ambas utilizadas para pesquisa e desenvolvimento de Inteligências artificiais todas elas criadas e evoluídas pela comunidade semeada por anos dentro da linguagem.

## 2.5 Legado de Guido Van Rossum

Guido não apenas criou o Python, mas deixou um impressionante legado para traz que transcende a própria linguagem, ele introduziu milhares de programadores a área da programação e inspirou milhões inclusive gerações e continua a moldar a forma como construímos um software hoje em dia mesmo tendo se passado décadas desde a criação do Python.

Fonte: Wikipedia

# 3 Compiladores

## 3.1 Papel Do Compiladores

Compiladores no contexto da programação desempenham um papel fundamental em seu ecossistema, são tradutores essenciais que transformam a linguagem humana em linguagem de máquina, eles traduzem o código fonte de uma linguagem de alto nível em código de máquina que é compreendido pelo hardware do computador, CPU, Memoria Principal entre outros periféricos intrínsecos ao sistema computacional.

## 3.2 CPython: Compilação JIT(Just-In-Time)

Todas as linguagens têm seus próprios compiladores e o Python não é diferente tendo o CPython como tradutor do seu código fonte ou linguagem compreendida pelo ser humano para a linguagem de máquina ou o bytecode, técnica que é chamada de compilação JIT (Just-in-Time), ele permite que a linguagem seja interpretada em diferentes sistemas operacionais como Windows, Linux entre outros a partir de uma compilação dinâmica, diferentemente de uma compilação estática, onde o todo código é traduzido antecipadamente ela traduz o código em simultâneo com a execução melhorando o desempenho do programa.

# 4 Exemplos Práticos Em Python

## 4.1 Sintaxe Simples, Mas Poderosa

No Python diferente de outras linguagens vemos a facilidade e eficácia de código que a linguagem nos proporciona aprendemos que apenas escrevendo **print("Olá Mundo")** já se pode exibir informações na tela enquanto outra linguagens como Java ocupam 4 a 5 linhas o Python necessita apenas de uma linha de comando para realizar o trabalho inteiro.

Figura 1: Programa Python

```
1 # file.py
2 print("Hello World")
```

Fonte: Ed Rodrigues

Figura 2: Programa em Java

```
public class HelloWorld{
  public static void main(String[] args) {
    System.out.println("Hello, world!")
  }
}
```

Fonte: Dev Community

## 4.2 Variáveis – Baús De Informações

As variáveis na programação são um dos se não o comando mais importante de linguagens como um todo, elas guardam qualquer informação de números, letras a caracteres especiais. São essenciais para qualquer programa mais complexo ser realizado, principalmente quando inputs são utilizados, eles coletam as informações, mas são apenas as variáveis que conseguem guardá-las e utilizá-las em um momento oportuno com o decorrer do programa.

## 4.3 Input – Comando De Leitura

O input é o principal comando do Python quando se trata do quesito interação com o usuário, ele permite que sejam escritas informações específicas para a execução do programa de forma dinâmica, algumas informações mais comumente pedidas são: nomes de usuário, idades e até salário se o programa se tratar de um cálculo de desconto ou acréscimo salarial dentre outras gamas de informações essenciais que o programa pode requirir para seu funcionamento.

## 4.4 Tuplas, Listas e Dicionários – Cadeias De Informações

As tuplas, listas e dicionários são um conceito mais avançado da linguagem Python, elas desempenham papel fundamental na manipulação e organização de informações, fornecem maneiras distintas de acessas dados cada uma com suas próprias características.

### 4.4.1 Tuplas

Tem como principal característica a imutabilidade, uma vez que o elemento for criado dentro da tupla ele passa a ser imutável. As tuplas são representadas pelos parênteses **‘( )’**.

Figura 3: Tupla

```
# Exemplo de tupla
ponto = (10, 20)
```

Fonte: ChatGPT

### 4.4.2 Listas

Tem como característica a mutabilidade e a versatilidade sendo a mais usada atualmente, elas permitem a adição, remoção e modificação de elementos já dentro da lista. São representadas pelos colchetes **‘[ ]’**

Figura 4: Listas

```
# Exemplo de lista
numeros = [1, 2, 3, 4, 5]
```

Fonte: ChatGPT

### 4.4.3 Dicionários

Tem como principal característica a associação de palavras-chave, cada valor dentro dos dicionários é acessado por sua palavra-chave correspondente, ele disponibiliza menos poluição visual e ajuda o programador a analisar melhor o programa, são úteis quando associados a

uma entidade específica como usuários, profissões etc. São representados pelas chaves **‘{ }’**

Figura 5: Dicionários

```
# Exemplo de dicionário
pessoa = {"nome": "João", "idade": 25, "profissao": "engenheiro"}
```

Fonte: ChatGPT

## 4.5 Comandos De Exceção

Python como todas as linguagens necessita das exceções que são: if, elif e else, eles executam os caminhos possíveis de um comando. Aprender esses “caminhos” é uma das principais habilidades que o programador deve ter e o Python simplifica esse conceito o que mais uma vez o torna uma das melhores linguagens para qualquer iniciante.

Figura 6: Inputs, If, elif e else

```
usuario = input("Informe o usuário do sistema!: ")

""" BLOCO DE INSTRUÇÃO COM ELIF"""
if usuario == "Flávio":
    print("Seja bem vindo Flávio")

elif usuario == "Douglas":
    print("Seja bem vindo Douglas")

elif usuario =="Nico":
    print("Seja bem vindo Nico")
else:
    print("Usuário não identificado!")
```

Fonte: Cursos Alura

## 4.6 Comandos De Repetição

Comandos de repetição são essências na programação, eles permitem que tarefas repetitivas sejam executadas de maneira eficiente evitando a execução manual desnecessária, principalmente quando estamos falando de uma análise extensa de dados como o uma lista de produtos de uma empresa de varejo que precisam ser catalogados ou até mesmo uma tarefa periódica que ocorre sempre no mesmo horário todos os dias como um envio de E-mails de marketing para contatos cadastrados em sua empresa, se esse processo for repetitivo e cansativo o Python pode resolve-lo com seus 2 principais comandos **‘for’** e o **‘while’**

### 4.6.1 Comando For

O comando **for**, que pode ser traduzido como “para”. É utilizado coleções de dados como listas, tuplas (variação de lista) e strings. Ele junta esses elementos dessas coleções e executa um bloco de código para cada elemento correspondente que está dentro destas, é útil quando o processo para cada elemento é repetitivo e cansativo. O uso de **‘range’** amplia a flexibilidade do **‘for’**, e permite que o comando seja executado em intervalos ou sequências numéricas.

Figura 7: For

```
1  for x in range(5):
2      print('Bom dia')
```

```
OUTPUT  ...  Code
Bom dia
Bom dia
Bom dia
Bom dia
Bom dia
```

Fonte: Autoria Própria

No exemplo acima o comando **“for”** executa uma sequência de 0 a 4 realizando um bloco de código que exibe na tela **“**Bom Dia**” 5** vezes.

### 4.6.2 Comando While

Seguindo o contexto de comando de repetição temos agora o comando While ou em tradução geral “enquanto”. É valioso para identificar e executar um bloco de código quando condição que foi imposta a ele for verdadeira, é necessário em situações que a quantidade de vezes que o programa será repetido não é conhecida antecipadamente.

Figura 8: While

```
# Exemplo de uso do comando while
contador = 0
while contador < 5:  # Executa enquanto contador for menor que 5
    print(contador)
    contador += 1
```

Fonte: ChatGPT

# 5 Bibliotecas Variadas

## 5.1 Desenvolvimento Web Com Django

Python é uma linguagem enorme, principalmente no que já foi falado sobre a comunidade que sempre cresce com novos programadores a todo momento aprendendo sobre a linguagem, e foi em 2003 que Django um dos frameworks ou bibliotecas mais fortes do Python no quesito desenvolvimento web surgiu. Foi desenvolvido por Adrian Holovaty e Simon Willison que tinham o objetivo de criar uma biblioteca que fizesse o programador não se preocupar tanto com as configurações da infraestrutura do site como o banco de dados e se preocupar mais com a lógica de programação que necessita de mais raciocínio e tempo do programador

## 5.2 Automações Web com Selenium

Criada por Jason Huggins em 2004 **Selenium** é uma biblioteca poderosa que permite o programador navegar na web de forma automatizada, já muito famosa entre os testadores de softwares essa biblioteca permite criar automações interativas em um navegador web, o que garante a integridade da aplicação testando (se bem programada) todos os possíveis erros que podem ocorrer no acesso do usuário real. Além dessa função Selenium pode ser utilizado para Scraping de dados ou extração de dados de uma página web, se for uma página de finanças pode ser útil para tirar informações necessárias para uma tomada de decisão mais consciente de uma empresa de negócios ou até um resumo do conteúdo de um site ou vários sites.

## 5.3 Automações com PyAutoGui

Esta é mais uma biblioteca extremamente poderosa que automatiza o seu teclado, basicamente você pode programar o seu

teclado para agir sem que você toque nele facilitando o processo em tarefas repetitivas que uma pessoa comum pode ter utilizando um computador como um cadastro de produtos em uma tabela Excel, para criar um programa eficiente com essa biblioteca basta ter a lista dos produtos em algum arquivo e programar quais lugares deverão ser preenchidos com seus nomes, Basta utilizar os fundamentos dos comandos de repetição e o passo a passo de sua tarefa e essa biblioteca se torna tão eficiente quanto qualquer funcionário remunerado no mercado de trabalho, podendo ainda ser programada a tarefa para ser realizada periodicamente com o Gerenciador de Tarefas do Windows.

## 3 CONCLUSÃO

Desde sempre a programação vem tendo um grande problema, muitas pessoas abandonam a área da programação antes mesmo de começarem devido à falta de motivação.

De Acordo com Pedrosa (2023):

> A evasão nos cursos de graduação da área de Tecnologia da Informação (TI) é maior que a média de evasão de todas as áreas de ensino. De acordo com dados do Mapa do Ensino Superior no Brasil, divulgados recentemente, a taxa de evasão nos cursos presenciais de TI, em 2021, foi de 38,5%, frente uma média de 30,7% das demais áreas. Nos cursos de ensino à distância (EAD), a taxa foi de 38%, contra 36,5% nas demais áreas. Por outro lado, o setor de computação e tecnologia é um dos que mais demandam mão de obra especializada no país.

meio contraditório quando sabemos que TI é uma das áreas que mais cresce a demanda atualmente.

De acordo com Dino (2023):

> Procura por profissionais de TI será de 420 mil pessoas até 2024 e o Brasil forma apenas 46 mil por ano (DINO, 2023, ONLINE)

por que tão poucas pessoas se formam? Uma das respostas para essa pergunta são a falta de motivação, a extrema luta psicológica que um programador trava consigo mesmo ao encarar problemas, a gigante gama de áreas que existem que são extremamente extensas em conteúdo e que assustam qualquer iniciante.

E o Python nasce para resolver o problema, ele ameniza o processo de aprendizado principalmente da lógica de programação para o profissional iniciante sem excesso de códigos tendo uma tipagem extremamente dinâmica e legível.

Suas bibliotecas, o grande forte da linguagem, economizam muitas vezes trabalho e raciocínio do programador. Django por exemplo, excelente ferramenta de desenvolvimento Web que permite o programador utilizar banco de dados sem tocar em MySql, reutilizar e economizar códigos de maneira incrivelmente simples e usar do próprio Python em Desenvolvimento Web o que por si só aumenta a gama de

possibilidades devido ao poder de juntar suas variadas bibliotecas diferentes em um projeto apenas.

E existem centenas de formas de aprender a linguagem, desde sozinho em casa com livros tanto físicos quanto online, vídeo aula e até na própria documentação da linguagem que fica disponível na internet ou com ajuda externa de professores como cursos técnicos profissionalizantes, faculdade entre outros. Python é uma linguagem imponente no mercado, divisora de águas no setor atualmente e quem a domina tem o futuro nas mãos, mas o conhecimento só se internalizará se for externalizado de alguma forma, por isso resolução de exercícios, testes e muita prática são essenciais no processo de aprendizado, mesmo que a linguagem seja fácil sem força de vontade para praticar e exercitar sua mente nem o comando mais básico será aprendido e a única coisa que voltará será o desanimo e o tempo perdido.

## GLOSSÁRIO

Sintaxe:

É a regra gramatical específica a formações de frases ou expressões em uma linguagem de programação

Machine Learning:

Machine Learning é o aprendizado de máquina, a parte da programação que cria as inteligências artificias

Back-end:

Back-end é a parte lógica da programação, onde as próprias linguagens de programação estão mais concentradas

Framework:

São um conjunto de bibliotecas que fornecem comandos reutilizáveis para desenvolvedores, usados para otimizar a construção de sites

# REFERÊNCIAS

COLÉGIO INTEGRADO. **Procura por profissionais de TI será de 420 mil pessoas até 2024 e o Brasil forma apenas 46 mil por ano**: Como os jovens que almejam seguir carreira na área tecnológica estão se preparando. [*S. l.*], 13 jan. 2021. Disponível em: https://g1.globo.com/go/goias/especial-publicitario/colegio-integrado/o-futuro-e-integrado/noticia/2021/01/13/procura-por-profissionais-de-ti-sera-de-420-mil-pessoas-ate-2024-e-o-brasil-forma-apenas-46-mil-por-ano.ghtml. Acesso em: 3 mar. 2024.

. **Estudantes de TI têm alto índice de abandono da universidade**: Cerca de 60% dos alunos dos cursos como ciência da computação, design de games ou sistema de informação não se formam. [*S. l.*], 14 ago. 2023. Disponível em: https://g1.globo.com/go/goias/especial-publicitario/colegio-integrado/o-futuro-e-integrado/noticia/2021/01/13/procura-por-profissionais-de-ti-sera-de-420-mil-pessoas-ate-2024-e-o-brasil-forma-apenas-46-mil-por-ano.ghtml. Acesso em: 3 mar. 2024.

PEDROSA, Raissa. **Quatro em cada dez alunos das faculdades de TI abandonam curso; entenda por quê**: Por um lado, mercado pede cada vez mais profissionais e contrata mesmo sem formação, por outro, universidades defendem ensino superior como forma de ampliar horizontes. [*S. l.*], 31 jul. 2023. Disponível em: https://www.otempo.com.br/economia/quatro-em-cada-dez-alunos-das-faculdades-de-ti-abandonam-curso-entenda-por-que-1.3093736. Acesso em: 3 mar. 2024.

PREVOT, Vanessa. Linguagem de Programação I Python. **Técnico em Informática**, Colégio Realengo, p. 20-50, 29 fev. 2020. Disponível em: https://biblioteca.colegiorealengo.br/leitura/pdf/4/1598997103-LING%20DE%20PROGRAMA%C3%87%C3%83O%20I%20%20-%20Livro_compressed.pdf. Acesso em: 3 mar. 2024.

FIGUEIRA, Vinícius. **Desenvolvimento Web com Python e Django**. Python Academy: [*s. n.*], 2020?. 130 p. v. 1. Disponível em: https://www.academia.edu/40203959/Desenvolvimento_web_com_python_e_django. Acesso em: 3 mar. 2024.